Image Processing – Prediction Task

1. Cat or Dog?

In this challenge, you need to predict whether an image contains either a cat or a dog.

This is useful for tasks such as animal species image recognition for restricting access. Even though a human being can easily differentiate between a cat and a dog, it is a tedious task that needs full-time availability. We can use machine learning to find an automated solution for that.

You can use any machine learning algorithm to build the models, either from the sessions or your own research based on the topics we discussed.

Note: Use 100 as the value for any random state you set to generate comparative results.

2. Data set

Within the data folder, there are two folders and two .csv files which are further described below.

train_images folder

Folder which contains the labelled images. Each image has an id as its name which is referred to by the "train.csv" file.
Number of image files: 10,000
File format: .jpg

train.csv file

Details of training data to use with model training and validation.
Number of entries: 10,000
Columns:

- id – unique id assigned to each image in the train_images folder
- label – image label (cat or dog)

test_images folder

Folder which contains the test images. For these images, labels need to be predicted using the built model.

Number of image files: 1,000
File format: .jpg

test.csv file

Testing data to use with predictions.
Number of entries: 1,000
Columns:

- id – unique id assigned to each image in the test_images folder

3. Submission

The final submission should include the following:

a. Test data predictions

Your test predictions need to be submitted to the **CodaLab**. Please carefully read the submission instructions given on the CodaLab page before submission.

Make sure you use the **best** model(s) you built to make predictions on the test dataset.

b. Blog explaining your implementation

A maximum 300-word blog explaining what you did and why including any plots and screenshots of the code needs to be submitted to **Moodle**.

Make sure to include evaluation results that you use to compare the models you built to select the best model in your blog.

Note: Both elements mentioned above **must be** submitted to consider your submission as a valid submission.

Hints:

Model evaluation

Use 70% of the training data to train the machine learning model and the remaining 30% as the validation set to evaluate the model.

The following code segment can be used to perform data splitting.

```
X_train, X_val, y_train, y_val = train_test_split(X, y, test_size=0.3,
random_state=100)
```

Save Predictions

The following code allows saving the predictions into a .csv file. Make sure to fill in the missing parts in this code according to your training process and use the correct model to make predictions.

```
# read csv file into a dataframe
df_test = pd.read_csv('/content/test.csv')

# load test images and resize them to the base size used with training
images
<add your code here>

features_list = []
# get features for test images following the same procedure used with
training images
<add your code here>

# make predictions using the best model
predictions = model.predict(features_list)

# add predictions to the 'Label' column
df_test['prediction'] = predictions

# save data frame to .csv file
df_test.to_csv('/content/test-predictions.csv', index=False)
```

The saved .csv file can be used to generate the final submission file using the Python scripts provided with the CodaLab page.